NGST
Next Generation Space Telescope

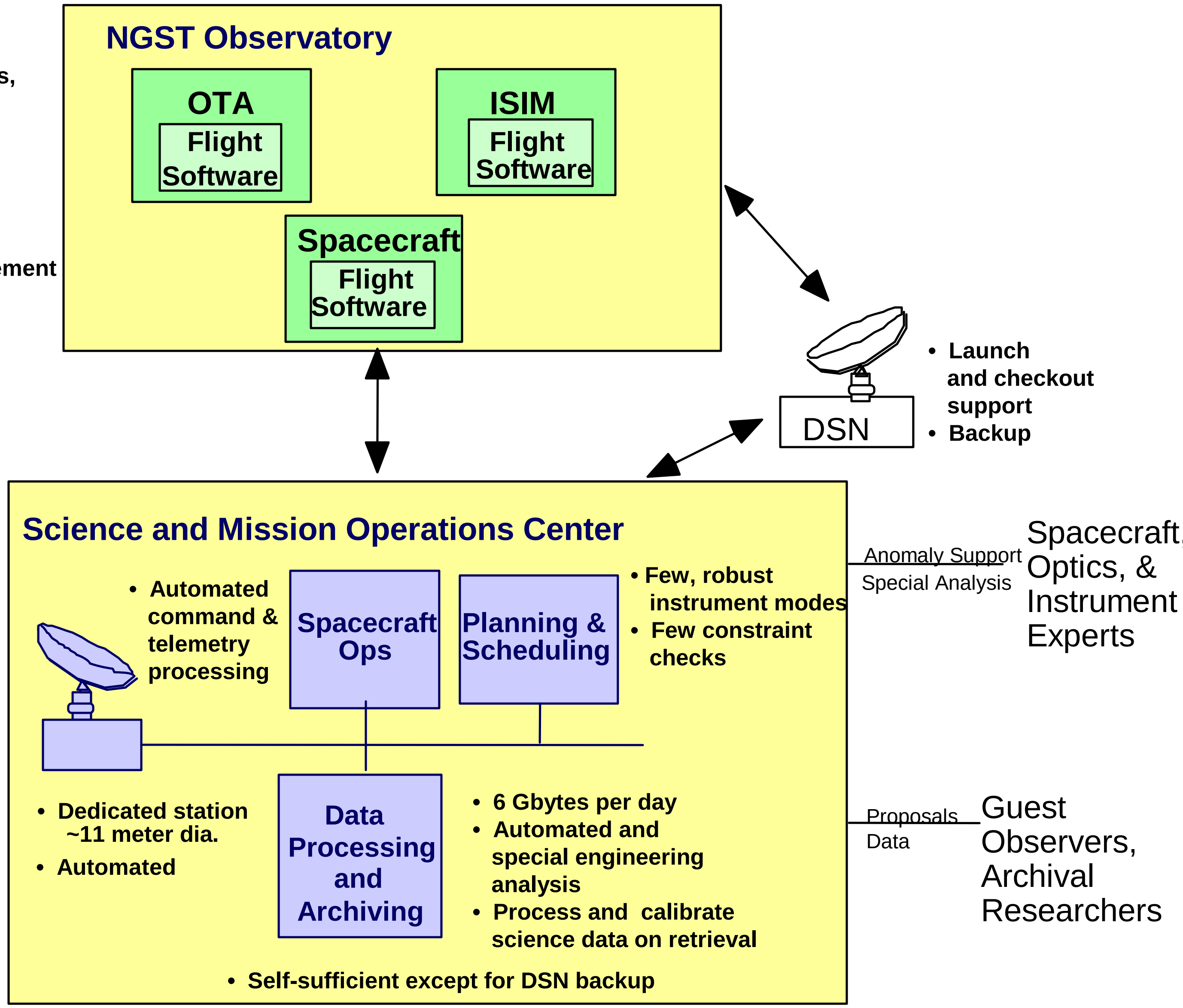
NGST TIMELINE EXECUTION AND DATA SYSTEM CONCEPTS

J. Keith Kalinowski
kkalinowski@hst.nasa.gov

Goddard Space Flight Center

Baseline NGST Operations Overview

- Common software components, standards
- File to file transfers of loads, observation data
- Simple, robust safemode
- Onboard autonomy
  - attitude
  - guide star selection
  - activity sequencing
  - momentum management
  - antenna pointing
  - telemetry filtering



- All functions co-located
  - cross-trained ops
  - common data bases
  - single cmd generator
  - single archive
  - reduced comm costs
  - no contention for station

Next Generation Space Telescope

NGST

A NASA Origins Mission

443.041

Event-driven operations

- **Definition of concepts**
  - **Time-tagged - sequence at specific or relative time**
  - **Event-driven - sequence of onboard activities driven by current status**
    - **Time *can* be an event**

Event-Driven vs. Time-Driven Operations: Flexibility

- Event-driven operations are more robust
    - can more flexibly adapt to operational realities and changes in capabilities and performance over time
- Event-driven operations naturally accommodate time-variant operations modes
    - for example, an exposure meter mode
- Event-driven operations should reduce the amount of operator effort in special operations
    - commissioning, safemode, rescheduling
- Event-driven operations enable autonomy
    - decisions on next activity based on results of previous activity

Event-Driven vs. Time-Tagged Operations

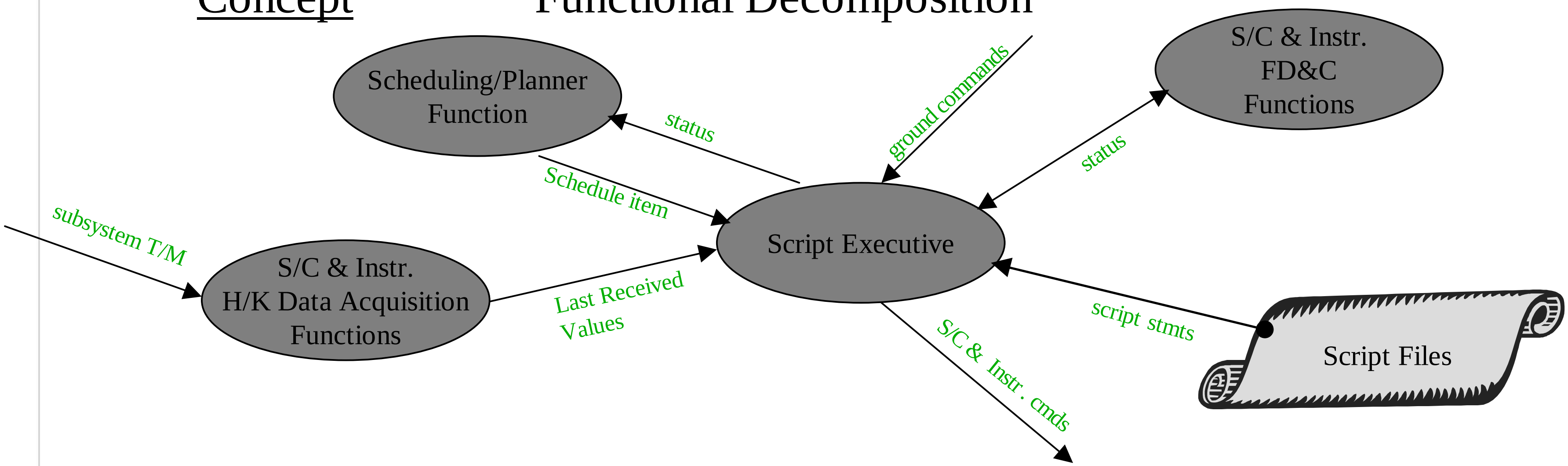
- Event-driven operations should increase efficiency 2% to 3%
  - Immediately move on to next observation when an observation fails
  - Sequence activities based on actual status, rather than predictions or worst-case estimates
    - Reduce time for attitude maneuvers and settling, instrument configuration
  - Reduce frequency of momentum dumps - performed when required, rather than routinely

Miscellaneous Aspects of Event-driven Operations

- **Analysis shows that event driven software does not have tight software timing constraints or drive onboard processor or memory requirements**
- **Onboard systems must either report events or generate data that event software can convert to events; for example:**
  - **target acquisition from the instrument**
  - **wheel speeds from the momentum wheels**
- **Real-time commanding may require coordination with onboard event software**
- **Data accounting needs as-flown timeline reported by onboard system**
- **Traditional command management function can be much simpler**
  - **many ground models of onboard performance not needed**
  - **remaining models require lower fidelity**

Flight Software Concepts
Adaptive Scheduling (AS)

Concept Functional Decomposition



Current Activities

- In-house flight implementation of Script Executive is on an 18-month development schedule
- A study of specific NGST scheduler/planner functionality has been initiated

Flight System Autonomy And Automation

- **Definition - onboard decision making and processing**
- **Rationale - lower cost, improve efficiency, lower risk**
- **Candidate autonomous functions**
  - **catalog of 47 or so, ranging from "state-of-the-practice" to unproven**
- **Process**
  - **define benefits and costs**
  - **estimate maturity without NGST investment**
  - **identify potential partners**
  - **invest in technology with results by 2003**

NGST Flight Autonomy and Automation

Included in current NGST Concept

State of the Practice	Baselined for NGST	Possible for NGST	Not Needed for NGST
- Onboard data management - Attitude determination - Antenna pointing - Thermal management - Power management - Limit checking of telemetry - Spacecraft mode management (e.g., safe mode entry) - Redundancy management	- Event driven management of activities/Adaptive scheduling - Target acquisition - Instrument direction of spacecraft activities - Momentum dumping - Vibration control	• Instrument exposure termination based on signal • Autonomous mirror adjustments/calibration • Spacecraft state modeling (for fault detection) • Selected error recovery • Science processing (for example, cosmic ray removal)	• Goal driven Scheduling • Stationkeeping • Fault diagnosis • Safemode recovery • Adaptable calibration

NGST Flight Data System Requirements

Collect focal plane array data and build compressed (~3:1 lossless) files.

Software constructs each exposure (with a duration of 1000 seconds or greater) by reading pixels from Focal Plane Arrays (FPAs) and either manipulating multiple pre- and post-integration readouts or "sampling up the ramp" and performing a slope fit. Each observation file contains a series of exposures for a given observation.

Provide on-board data storage for one day's worth of science data files.

All files are transferred to the ground once per day during a continuous 8 to 13 hour contact. Each file is deleted from the on-board archive once it is has been successfully transferred to the ground.

Support S/C to ground communications (including file transfer).

Real-time CCSDS commands as well as "file loads" are transmitted to the spacecraft. Real-time engineering telemetry is transmitted to the ground in CCSDS format. Science (and engineering) data files are transferred to the ground via a standard file transfer protocol.

Execute attitude determination and control algorithms.

Gyro processing at 10Hz and star tracker processing at 2Hz. ACS requirements are 2 arcsecond control accuracy and 0.3 arcsecond knowledge accuracy.

Control reaction control subsystem for momentum dumping and station keeping.

Control fast steering mirror at 100 Hz.

Provide a telescope pointing accuracy of 2 milliarcseconds by tracking a guide star falling on the focal plane array. The guide star is acquired autonomously. A small portion (100 pixels by 100 pixels) of an FPA (it can be any portion of any FPA) must be read at 100 Hz for use in controlling the Fast Steering Mirror. A 100 by 100 pixel field of view is a field of view of 2.9 by 2.9 arc-sec.

Support autonomous wavefront processing based on focal plane array data.

Image data is analyzed in a "open-loop" fashion (no timing requirements) and deformable mirror optics are adjusted.

Collect telemetry from subsystems and distribute commands to subsystems.

Control "Mirror Actuators."

Provide on-board health and safety monitoring of spacecraft and instrument components.

Autonomous detection of spacecraft and instrument failures and anomalies. For mission-critical failures, the data system will attempt to correct the problem. Non-mission-critical failures (or anomalies) are logged and reported to the ground.

Provide "event driven" (adaptive) mission timeline execution.

NGST Flight Data System Requirements

Requirements Analysis addressed the following areas:

Attitude Control System Requirements

- **10 Hz and 2 Hz control loops.**

Command and Data Handling System Requirements

- **Standard CADH capabilities with addition of on-board file storage and flight/ground file transfer.**

Instrument Data Collection Requirements

- **Focal Plane Array (FPA) data collection and processing.**

Fast Steering Mirror Control

- **Autonomous guide star acquisition and tracking using FPA data.**

Scheduling/Health & Safety Monitoring

- **Event-driven timeline execution**

NGST Flight Data System - Block Diagram

Functions of Data System Elements

The Spacecraft (versus Sciencecraft) Data System design is fairly traditional. The only unique requirements imposed on the spacecraft portion of the data system are the inclusion of the flight/ground file transfer capability and the high-speed network for communication with the ISIM components. This network is assumed to be capable of supporting TCP/IP using some standard physical layer (such as ethernet or ATM). Only "status information" transfers and file transfers are performed over this network; all tight real-time requirements are handled over more traditional spacecraft data interfaces. For purposes of this study, a "hot" redundant Spacecraft Data System processor is assumed. If this backup processor must take control, the ACS control mode will switch to a safe mode and the Mission Control Processor informed in order to coordinate the remaining spacecraft and instrument activities. Note that the same capability could also be achieved by placing a separate small processor in the ACS subsystem that supports a safemode independently of the Spacecraft Data System processor, but this has the downside of complicating communications with the Mission Control Processor if the S/C processor goes down.

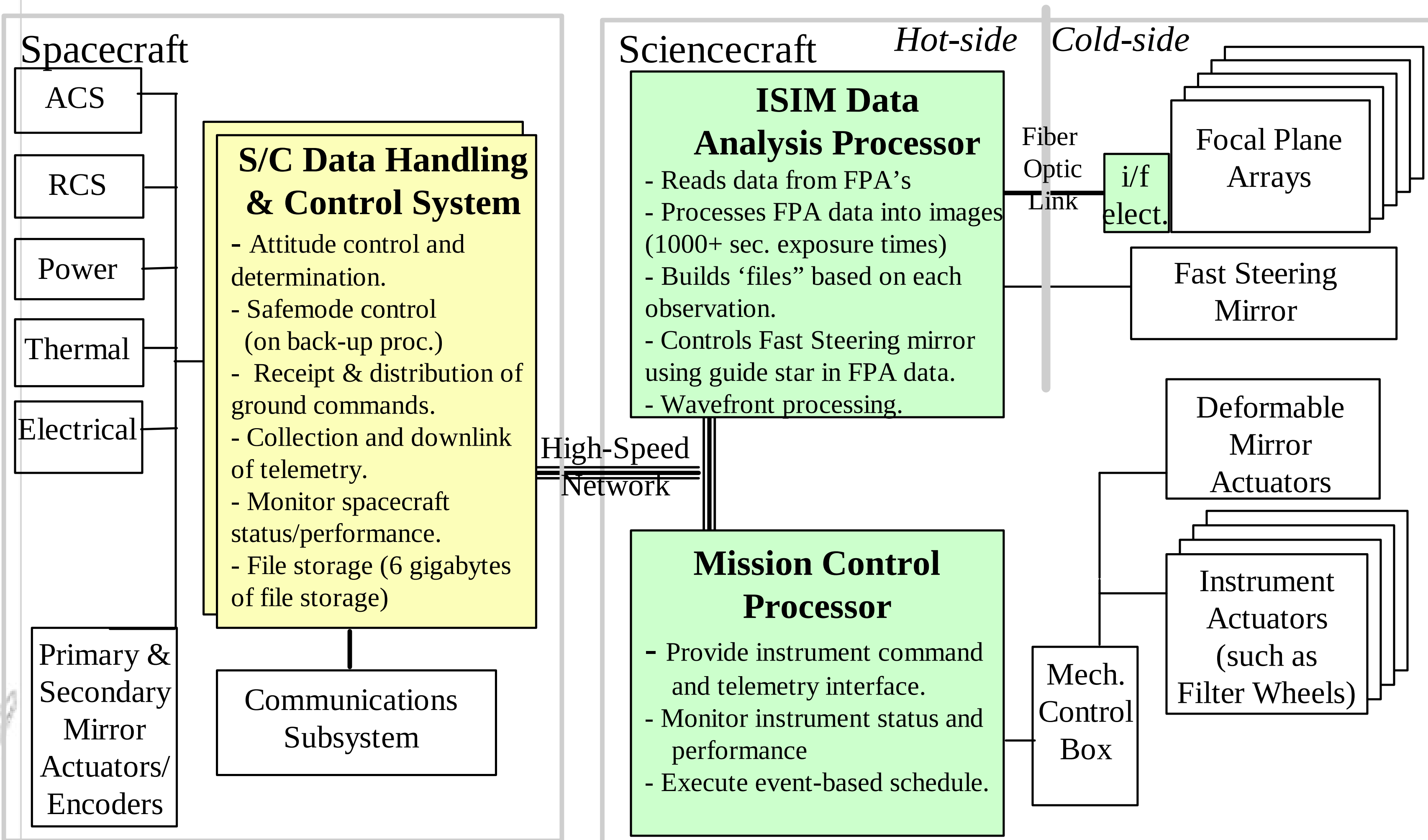
The ISIM Data Analysis Processor is responsible for reading the raw Focal Plane Array data from the FPA electronics, processing this data into "exposures" (either by manipulating multiple pre- and post-integration readouts or with "sample up the ramp slope fitting algorithms), and storing all of the exposures for a given observation into a file. Each observation file is compressed and transferred to the daily storage (via FTP) in the S/C Data System processor. Additionally, the ISIM Data Analysis Processor is responsible for controlling the Fast Steering Mirror at 100 Hz using a guide star identified within the FPA data. Lastly, this processor performs the wavefront processing needed to adjust the Deformable Mirror. This is not a "hard real-time" activity (it may take minutes) and note that the actual commands to the Deformable Mirror actuators are issued from the Mission Control Processor. One could move the wavefront processing to the Mission Control Processor - but the current architecture places all of the computationally intensive activities on the same processor based on the assumption that it may need to be optimized for this sort of processing.

The Mission Control Processor serves as the central coordinator for the NGST. It handles the scheduling of science activities as well as the control of the instrument components.

Goddard Space Flight Center

NGST Flight Data System - Block Diagram

Next Generation Space Telescope



NGST

A NASA Origins Mission

443.041

NGST Flight Data System Technology Assumptions

Flight Hardware Capabilities

- Rad-hard version of a commercial processor which is at least as powerful as 150 MHz PowerPC 740 (a rad-hard PowerPC 740 should be available within the next 3 years)
- High-speed on-board TCP/IP network (ethernet or ATM) for interprocessor communications (need ~10 Mbps capability; work on much higher rates is underway)
- Dynamic RAM with EDAC for large quantities of on-board storage is available now

Software Capabilities

- Commercial Real Time Operating System (RTOS), such as VxWORKS, for tasking, file management, and networking support (available now)
- Availability of a standard mechanism for satellite/ground file transfer (work is underway)
- Spacecraft processors, networks, and buses that are "COTS-like" will allow the use of the standard drivers for these devices.

These technologies should be available for spaceflight in the NGST timeframe.

Benefits of using COTS-like Data System Hardware

The use of flight hardware that is identical in function to commercially available hardware has many advantages during development. For example, if a commonly available microprocessor such as a PowerPC is used, a number of development tools (such as compilers and in-circuit emulators) are available to aid in the development. Additionally, a commonly available hardware platform ensures that most commercially available software (including real-time operating systems such as VRTX and VxWORKS) will be available for the platform.

When configuring a flight software test system, if a standard processor and bus is used for flight (such as a PowerPC and a CompactPCI bus), commercially available boards can be used to set up a flight software test environment by purchasing them and installing them into a Compact PCI chassis. This reduces the cost of the test environments AND allows flight software development to get started in parallel with the design of the actual flight hardware. Note that the concept here is that the flight hardware is identical in FUNCTION to commercially available hardware, but it is not necessarily exactly the same design. (This approach does not mean that the flight software does not have to eventually be tested on the real flight hardware.)

Finally, interface testing is simplified for a couple of reasons. First, if the communications link between processors uses a standard software protocol, such as TCP/IP, it is possible that some interface testing could occur remotely. Additionally, it is also possible to build "simple interface test sets" using commercial hardware along with the actual flight software. In the NGST case, this would be invaluable to the Focal Plane Array integrators as each could be provided an interface simulator (based on real flight software) that allows them to read data from their FPA and transfer it (via TCP/IP) to a local workstation.

Benefits of using COTS-like Data System Hardware

Use of flight hardware that is identical (in function) to commercially available hardware enables lower system development costs because it...

- **Enables the use of commercial software development tools for flight software development**
- **Enables the use of commercial software and commercial software standards**
- **Lowers the cost of flight software test systems, because the commercial equivalent to the flight hardware can often be used**
- **Simplifies interface testing (such as enabling the development of low-cost interface test sets)**

Protocols and Formats

- Traditional technique
  - CCSDS packets, level zero processing
- NGST Concept
  - Observation files, file transfers

Traditional	Files/File Transfers
Requires LZP on ground	Onboard files could go directly to archive
Tools and techniques established	Requires establishment of standards
	Requires uplink bandwidth

- Standards exist or are under development
  - FITS file format - imposes some overhead on data
  - CCSDS standards under development

	Commercial	CFDP	SCPS
Description	Adapt ftp protocol to NGST environment (e.g., 5 second latency	Under development by CCSDS	Developed by CCSDS
Uplink	require ~40 kbps	modest bandwidth required	modest bandwidth required
Complexity	high	medium	high
Requirements satisfaction	meets	meets	meets
Potential for widely accepted standard	Low	Medium	Low